文章
姚 鑫 · 五月 18, 2021 阅读大约需 6 分钟

第四章 使用Setup和tear Down方法执行测试

第四章 使用Setup和tear Down方法执行测试

示例：使用Setup和tear Down方法执行测试

以通常的方式执行新的单元测试。

1. 在一直在使用的命名空间中打开终端。
2. 将^UnitTestRoot的值设置为包含测试类的目录的父级：

```
USER> Set ^UnitTestRoot="c:\unittests"
```

3. 使用%UnitTest.Manager执行测试：

```
USER> Do ##class(%UnitTest.Manager).RunTest("mytests")
```

4. IRIS加载测试类、编译类、执行测试并向终端发送报告。

```
================================================================================
Directory: C:\unittests\mytests\cls\MyPackage\
================================================================================
mytests\cls\MyPackage begins ...
Load of directory started on 01/09/2018 14:36:57 '*.xml;*.XML;*.cls;*.mac;*.int;*.inc
;*.CLS;*.MAC;*.INT;*.INC'

Loading file C:\unittests\mytests\cls\MyPackage\Tests.xml as xml
Imported class: MyPackage.Tests

Compilation started on 01/09/2018 15:44:01 with qualifiers ''
Compiling class MyPackage.Tests
Compiling routine MyPackage.Tests.1
Compilation finished successfully in 0.033s.

Load finished successfully.

  MyPackage.Tests begins ...
      TestAdd() begins ...
        AssertEquals:Test Add(2,2)=4 (passed)
        AssertNotEquals:Test Add(2,2)'=5 (passed)
        LogMessage:Duration of execution: .000073 sec.
      TestAdd passed
      TestEditContact() begins ...
        AssertStatusNotOK:ContactType = Friend (passed)
        AssertStatusOK:ContactType = Personal (passed)
        LogMessage:Duration of execution: .001227 sec.
      TestEditContact passed
```

```
    MyPackage.Tests passed
  mytests\cls\MyPackage passed

Use the following URL to view the result:
http://10.0.75.1:52773/csp/sys/%25UnitTest.Portal.Indices.cls?Index=10&$NAMESPACE=USE
R
All PASSED
```

执行测试的选项：测试规格和限定符

通常，可以使用以下形式的命令执行RunTest：

```
Do ##class(%UnitTest.Manager).RunTest("testspec","qualifiers")
```

Testspec参数确定要运行哪些测试以及在哪里可以找到它们。Testspec的一般形式是testSuite：testcase：testmeth od，其中
- testsuite(必填)。包含导出的测试类的文件目录。该目录必须是名为^UnitTestRoot的目录的子目录。默认情况下，测试管理器执行此目录及其子目录中包含的所有文件中的所有测试。
- testcase测试用例(可选)。选择包含要执行的测试方法的单个类。格式为PackageName.ClassName。如果存在，则测试管理器仅执行命名类中的测试。
- testmethod(可选)。挑选由测试用例指示的测试类的一个方法来执行。

限定符参数指定用于运行测试的各种选项。正如我们已经看到的，当想要从.cls文件加载测试时，可以使用 “ /loadudl ” 限定符。还可以使用限定符来控制测试类在执行后是否从服务器中删除，是否应该从这些外部文件加载测试，或者系统是否应该在测试失败后进入调试模式，等等。限定符参数是一个可选的命令行参数字符串，用于打开或关闭某些测试管理器行为。例如，“ /NoLoad/DEBUG ”告诉管理器不要从目录加载任何测试，也就是说，使用当前在InterSystem s IRIS中的测试，并在调试模式下运行测试。这些限定符就是所谓的可否定布尔值。例如，这意味着 “ /NoLoad ” 等同于 “ /Load=0 ” 。

限定符	含义
/load (default)	从目录加载测试。使用/NoLoad不加载测试，并执行InterSystems IRIS中已包含的测试。
/run (default)	运行测试。使用/norun加载但不运行任何测试。
/delete (default)	执行后从InterSystems IRIS中删除测试类。使用/nodelete保存类。
/recursive (default)	在指定目录的子目录中查找测试。使用/norecsive不执行子目录中包含的测试。
/debug (default is /nodebug)	使用/DEBUG，第一次测试失败后不会执行任何测试。从终端执行时，终端将在第一次故障后进入调试模式。
/autoload	使用/autoload=dir从^UnitTestRoot目录的子目录 “ dir ” 加载测试。
/loadudl	从.cls而不是XML文件加载测试。

RunTest 示例

以下是使用RunTest执行单元测试的一些示例。

要使用RunTest，必须首先为^UnitTestRoot分配一个有效的目录名：

```
USER>Set ^UnitTestRoot = "C:\UnitTests"
```

例1：

```
USER>Do ##class(%UnitTest.Manager).RunTest()
```

在^UnitTestRoot目录的所有子目录中搜索包含测试类的XML文件。加载它找到的任何测试类并执行测试。

执行后从InterSystems IRIS中删除所有加载的测试类。

例2：

```
USER>Do ##class(%UnitTest.Manager).RunTest("mytests")
```

- 加载并执行^UnitTestRoot的mytests子目录(及其子目录)中的测试。
- 在测试类执行后从InterSystems IRIS中删除它们。

例3：

```
USER>Do ##class(%UnitTest.Manager).RunTest("mytests:MyPackage.Tests")
```

- 从^UnitTestRoot目录的mytest子目录(及其子目录)加载测试。仅执行MyPackage.Tests中的测试。
- 执行测试后从InterSystems IRIS中删除所有测试类。

例4：

```
USER>Do ##class(%UnitTest.Manager).RunTest("mytests:MyPackage.Tests", "/noload/nodele
te")
```

- 不将测试加载到IRIS。
- 在MyPackage.Tests中执行测试。请注意，mytest必须仍然包含带有MyPackage.Tests类的XML文件。
- 不从IRIS中删除MyPackage.Tests。

DebugRunTestCase

%UnitTest.Manager类还包含DebugRunTestCase方法。若要使用此方法，仍必须先将^UnitTestRoot分配给有效目录：

```
USER>Set ^UnitTestRoot="C:\UnitTests"
```

例如：

```
USER>Do ##class(%UnitTest.Manager).DebugRunTestCase("mytests","MyPackage.Tests","","")
```

- 该方法不从任何目录加载任何类，也不从InterSystems IRIS删除任何类。
- 该方法执行MyPackage.Tests中包含的测试。
- 可选的第三个参数用于限定符。
- 可选的第四个参数用于指定测试类中要执行的单个测试方法。
- 如果测试失败，该方法将继续执行其余的测试方法，但将在测试完成时中断。因此，如果从终端执行，则终端将进入调试模式。

注意：使用DebugRunTestCase时，mytest目录实际上不需要包含MyPackage.Tests。相比之下，RunTest总是要求要执行的测试包含在^UnitTestRoot的子目录中，即使在使用NoLoad ” 时也是如此。

练习

练习1：MyPackage.TestMe包含一个名为CreateContact的方法。此方法创建并返回Contact实例。它接受Name和ContactType值作为参数。创建一个测试以下内容的单元测试：
- 从CreateContact返回的Contact实例具有正确的Name值。
- 从CreateContact返回的Contact实例具有正确的ContactType值。
- CreateContact返回的Contact实例保存正确，即%Save返回OK状态。

练习2：MyPackage.Contact包含名为ByContactType的类查询。它返回具有ContactType指定值的所有Contact实例的ID值。将单元测试添加到MyPackages.Tests，用于测试以下各项：

- 该查询返回指定ContactType的正确ID值数量。为此，必须正确初始化数据库。
- 查询返回的每个ID值对应于一个具有指定ContactType值的联系人。

请注意，添加此测试不应破坏在完成教程正文中的示例时添加到MyPackage.Tests中的测试。因此，必须以正确的方式初始化和恢复数据库。

把答案发到评论上！！！ 或加群QQ 410039091 分享

源码

#SQL #Caché #Global Masters #InterSystems IRIS #InterSystems IRIS for Health

---

源URL: https://cn.community.intersystems.com/post/%E7%AC%AC%E5%9B%9B%E7%AB%A0-%E4%BD%BF%E7%94%A8setup%E5%92%8Ctear-down%E6%96%B9%E6%B3%95%E6%89%A7%E8%A1%8C%E6%B5%8B%E8%AF%95